Practitioner's Docket No. GR 98 P 1764

CHAPTER II

"Express Mail" mailing label number EL608557808US
Date of Deposit November 20, 2000

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

MICHAEL BURNS

TRANSMITTAL LETTER
TO THE UNITED STATES ELECTED OFFICE (EO/US)

(ENTRY INTO U.S. NATIONAL PHASE UNDER CHAPTER II)

INTERNATIONAL APPLICATION NO.	INTERNATIONAL FILING DATE	PRIORITY DATE
PCT/DE99/01365	06 May 1999	20 May 1998

TITLE OF INVENTION

METHOD AND ARRANGEMENT FOR THE COMPUTER-AIDED EXCHANGE OF CRYPTOGRAPHIC KEYS BETWEEN A FIRST COMPUTER UNIT AND A SECOND COMPUTER UNIT

APPLICANT
HORN, Günther et al.

Box PCT
Assistant Commissioner for Patents
Washington D.C. 20231
ATTENTION: EO/US

1. Applicant herewith submits to the United States Elected Office (EO/US) the following items under 35 U.S.C. 371:

   a. [X] This express request to immediately begin national examination procedures (35 U.S.C. 371(f)).

   b. [X] The U.S. National Fee (35 U.S.C. 371(c)(1)) and other fees (37 C.F.R. § 1.492) as indicated below:

09/700928
528 Rec'd PCT/PTO 20 NOV 2000

2. Fees

<table>
<tr><td>CLAIMS FEE</td><td>(1) FOR</td><td>(2) NUMBER FILED</td><td>(3) NUMBER EXTRA</td><td>(4) RATE</td><td>(5) CALCULATIONS</td></tr>
<tr><td>[ ]*</td><td>TOTAL CLAIMS</td><td>62 - 20 =</td><td>42</td><td>x $ =18.00</td><td>$756.00</td></tr>
<tr><td></td><td>INDEPENDENT CLAIMS</td><td>2 - 3 =</td><td></td><td>x $ =</td><td></td></tr>
<tr><td></td><td colspan="4">MULTIPLE DEPENDENT CLAIM(S) (if applicable) + $270.00</td><td></td></tr>
<tr><td>BASIC FEE**</td><td colspan="4">[ ] U.S. PTO WAS INTERNATIONAL PRELIMINARY EXAMINATION AUTHORITY
Where an International preliminary examination fee as set forth in § 1.482 has been paid on the international application to the U.S. PTO:
[ ] and the international preliminary examination report states that the criteria of novelty, inventive step (non-obviousness) and industrial activity, as defined in PCT Article 33(2) to (4) have been satisfied for all the claims presented in the application entering the national stage (37 CFR 1.492(a)(4)) $98.00
[ ] and the above requirements are not met (37 CFR 1.492(a)(1)) $720.00

[X] U.S. PTO WAS NOT INTERNATIONAL PRELIMINARY EXAMINATION AUTHORITY
Where no international preliminary examination fee as set forth in § 1.482 has been paid to the U.S. PTO, and payment of an international search fee as set forth in § 1.445(a)(2) to the U.S. PTO:
[ ] has been paid (37 CFR 1.492(a)(2)) $790.00
[ ] has not been paid (37 CFR 1.492(a)(3)) $1,070.00
[X] where a search report on the international application has been prepared by the European Patent Office or the Japanese Patent Office (37 CFR 1.492(a)(5)) .... $860.00</td><td></td></tr>
<tr><td></td><td colspan="4">Total of above Calculations</td><td>= $1616.00</td></tr>
<tr><td>SMALL ENTITY</td><td colspan="4">Reduction by ½ for filing by small entity, if applicable. Affidavit must be filed. (note 37 CFR 1.9, 1.27, 1.28)</td><td>-</td></tr>
<tr><td></td><td colspan="4">Subtotal</td><td>$1616.00</td></tr>
<tr><td></td><td colspan="4">Total National Fee</td><td>$1616.00</td></tr>
<tr><td></td><td colspan="4">Fee for recording the enclosed assignment document $40.00 (37 CFR 1.21(h)). (See Item 13 below). See attached "ASSIGNMENT COVER SHEET".</td><td></td></tr>
<tr><td>TOTAL</td><td colspan="4">Total Fees enclosed</td><td>$1616.00</td></tr>
</table>

*See attached Preliminary Amendment Reducing the Number of Claims.

i. [X] A check in the amount of $1616.00 to cover the above fees is enclosed.

ii. [ ] Please charge Account No. _______ in the amount of $________.
A duplicate copy of this sheet is enclosed.

3. [X] A copy of the International application as filed (35 U.S.C. 371(c)(2)):

a. [X] is transmitted herewith.

b. [ ] is not required, as the application was filed with the United States Receiving Office.

c. [ ] has been transmitted

i. [ ] by the International Bureau.
Date of mailing of the application (from form PCT/IB/308):_______.

ii. [ ] by applicant on ____________.
Date

4. [ ] A translation of the International application into the English language (35 U.S.C. 371(c)(2)):

a. [ ] is transmitted herewith.

b. [ ] is not required as the application was filed in English.

c. [ ] was previously transmitted by applicant on ______________.
Date

d. [X] will follow.

5. [ ] Amendments to the claims of the International application under PCT Article 19 (35 U.S.C. 371(c)(3)):

a. [ ] are transmitted herewith.

b. [ ] have been transmitted

i. [ ] by the International Bureau.
Date of mailing of the amendment (from form PCT/IB/308):_______ _______.

ii. [ ] by applicant on ____________.
Date

c. [ ] have not been transmitted as

i. [ ] applicant chose not to make amendments under PCT Article 19.
Date of mailing of Search Report (from form PCT/ISA/210):_______ _______.

ii. [ ] the time limit for the submission of amendments has not yet expired. The amendments or a statement that amendments have not been made will be transmitted before the expiration of the time limit under PCT Rule 46.1.

6. [ ] A translation of the amendments to the claims under PCT Article 19 (38 U.S.C. 371(c)(3)):

a. [ ] is transmitted herewith.

b. [ ] is not required as the amendments were made in the English language.

c. [ ] has not been transmitted for reasons indicated at point 5(c) above.

7. [X] A copy of the international examination report (PCT/IPEA/409)
   [X] is transmitted herewith.
   [ ] is not required as the application was filed with the United States Receiving Office.

8. [ ] Annex(es) to the international preliminary examination report
   a. [ ] is/are transmitted herewith.
   b. [ ] is/are not required as the application was filed with the United States Receiving Office.

9. [ ] A translation of the annexes to the international preliminary examination report
   a. [ ] is transmitted herewith.
   b. [ ] is not required as the annexes are in the English language.

10. [X] An oath or declaration of the inventor (35 U.S.C. 371(c)(4)) complying with 35 U.S.C. 115
   a. [ ] was previously submitted by applicant on ________.
   Date
   b. [X] is submitted herewith, and such oath or declaration
      i. [X] is attached to the application.
      ii. [ ] identifies the application and any amendments under PCT Article 19 that were transmitted as stated in points 3(b) or 3(c) and 5(b); and states that they were reviewed by the inventor as required by 37 C.F.R. 1.70.
      iii. [ ] will follow.

Other document(s) or information included:

11. [X] An International Search Report (PCT/ISA/210) or Declaration under PCT Article 17(2)(a):
   a. [X] is transmitted herewith.
   b. [ ] has been transmitted by the International Bureau.
   Date of mailing (from form PCT/IB/308): ________________.
   c. [ ] is not required, as the application was searched by the United States International Searching Authority.
   d. [ ] will be transmitted promptly upon request.
   e. [ ] has been submitted by applicant on ______________.
   Date

12. [ ] An Information Disclosure Statement under 37 C.F.R. 1.97 and 1.98:
   a. [ ] is transmitted herewith.
   Also transmitted herewith is/are:
   [ ] Form PTO-1449 (PTO/SB/08A and 08B).
   [ ] Copies of citations listed.
   b. [ ] will be transmitted within THREE MONTHS of the date of submission of requirements under 35 U.S.C. 371(c).
   c. [ ] was previously submitted by applicant on ________________.
   Date

09/700928

528 Rec'd PCT/PTO 20 NOV 2000

13. [ ] An assignment document is transmitted herewith for recording.

A separate [ ] "COVER SHEET FOR ASSIGNMENT (DOCUMENT) ACCOMPANYING NEW PATENT APPLICATION" or[ ] FORM PTO 1595 is also attached.

14. [X] Additional documents:
- a. [X] Copy of request (PCT/RO/101)
- b. [X] International Publication No. WO 99/60747
  - i. [ ] Specification, claims and drawing
  - ii. [X] Front page only
- c. [ ] Preliminary amendment (37 C.F.R. § 1.121)
- d. [ ] Other

15. [X] The above checked items are being transmitted
- a. [X] before 30 months from any claimed priority date.
- b. [ ] after 30 months.

SIGNATURE OF PRACTITIONER

WERNER H. STEMER
REG. NO. 34,956

Lerner and Greenberg, P.A.
P.O. Box 2480
Hollywood, Florida 33020-2480
Tel.: (954) 925-1100
Fax: (954) 925-1101

PTO/PCT Rec'd 29 MAR 2001 #3

09/700928

Docket No.: GR 98 P 1764

CERTIFICATION

I, the below named translator, hereby declare that: my name and post office address are as stated below; that I am knowledgeable in the English and German languages, and that I believe that the attached texts are true and complete translations of application numbers 198 22 795.7, filed May 20, 1998 and PCT/DE99/01365, filed May 6, 1999.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Hollywood, Florida

Christine Kahl

Christine Kahl

March 26, 2001

Lerner & Greenberg, P.A.
P.O. 2480
Hollywood, FL 33022-2480
Tel.: (954) 925-1100
Fax.: (954) 925-1101

PTO/PCT Rec'd 29 MAR 2001 #3

09/700928

Description

Method and arrangement for the computer-aided interchange of cryptographic keys between a first computer unit and a second computer unit

The invention relates to the computer-aided interchange of cryptographic keys between a first computer unit and a second computer unit.

Information technology systems are subject to various threats. Thus, by way of example, transmitted information can be tapped and modified by an unauthorized third party. A further threat during communication between two communication parties is that of a false identity of one communication party being feigned.

These and other threats are countered by various security mechanisms which are intended to protect the information technology system from the threats. One security mechanism used for safeguarding purposes is encryption of the transmitted data. To be able to encrypt the data in a communication link between two communication parties, steps which prepare the encryption first need to be taken before the actual data is transmitted. By way of example, the steps may involve the two communication parties agreeing to an encryption algorithm and, if appropriate, the common secret keys being declared.

The encryption security mechanism takes on particular significance in the case of mobile radio systems, since the transmitted data in these systems can be tapped by any third party without any particular additional effort.

This leads to the requirement for known security mechanisms to be selected and these security mechanisms

to be suitably combined, and also for communication protocols to be specified, such that they ensure the security of information technology systems.

Various asymmetric methods for the computer-aided interchange of cryptographic keys are known.

Asymmetric methods which are suitable for mobile radio systems are described in [1], [2], [3] and [4].

The method described in [1] relates expressly to local area networks and makes relatively high demands in terms of computing power on a computer unit of a communication party during the key interchange. Moreover, more transmission capacity is required in the method than in the method according to the invention, since the length of the messages is greater than in the case of the invention.

The method described in [2] has not implemented a few fundamental security aims. Explicit authentication of the network by the user is not achieved. Moreover, a key transmitted to the network by the user is not confirmed to the user by the network. There is also no assurance for the network that the key is fresh (up to date). A further disadvantage of this method is the restriction to the Rabin method in the implicit authentication of the key by the user. This restricts the method in terms of more flexible applicability. In addition, no security mechanism which ensures the incontestability of transmitted data is provided. This is a considerable disadvantage, in particular also for the preparation of incontestable charge accounts for a mobile radio system. The restriction of the method to the signature function used being the National Institute of Standards in Technology Signature Standard (NIST DSS) also restricts the method in its general applicability.

The method described in [3] has not implemented a fundamental security aim: explicit authentication of the user by the network is not achieved.

The method described in [4] is based on the assumption of the existence of common secret keys both between the user and the visited network and between the user and the home network before a protocol pass starts. This assumption is too restrictive for many instances of use.

In addition, [5] discloses a method for secure data interchange between a multiplicity of subscribers involving a certification authority. The protocol used for this method has a random number, an identity statement and also a public key and a session key. However, fundamental security aims are not implemented in this method.

In addition, [6] discloses a method for PC-PC communication involving a trust center.

[7] discloses a method in which a session key is produced using both a public key and a secret key and also using a random number. This session key is combined with a public key.

In addition, [8] describes a method in which a user unit identifies itself to a network unit. An authentication process then takes place between the user unit and the network unit using a hash function.

[9] discloses further secure communication protocols which nevertheless do not implement important fundamental security aims.

[10] discloses the practice of forming a first value in a first computer unit from a first random number using

a generating element of a finite group, and transmitting it to a second computer unit. In the second computer unit, a session key is formed by hash value formation for the first value, which is exponentiated using a secret network key. The session key is likewise formed in the first computer unit, but there by hash value formation for a public network key which is exponentiated using the first random number. In addition, a hash value for the session key is formed there and the hash value is digitally signed. The resultant signature term is transmitted to the second computer unit and is verified there.

The method described in [11] achieves the important security aims, but with a relatively high level of input in terms of computing power and transmission capacity.

Asymmetric methods are essentially based on two complexity theory problems, the problem of efficiently factorizing composed numbers and the discrete logarithm problem (DLP). The DLP is that, although exponentiation operations can be carried out efficiently in suitable computing structures, no efficient algorithms are known for the reversal of this operation, logarithmation.

By way of example, the finite groups referred to above are to be understood as being such computing structures. These groups are, for example, the multiplicative group of a finite body (e.g. multiplication modulo p, where p is a large prime number), or else so-called "elliptical curves". Elliptical curves are primarily of interest because they permit much shorter security parameters for the same level of security. This relates to the length of the public keys, to the length of the certificates, to the length of the messages to be interchanged during session key declaration and to the length of digital

signatures, which are each described below. The reason for this is that the logarithmation methods known for elliptical curves are much less efficient than those for finite bodies.

In this context, a large prime number means that the size of the prime number needs to be selected such that logarithmation is so complex that it cannot be performed in a reasonable time. In this context, reasonable means a period of time corresponding to the security policy over a number of years to decades, and longer.

In this context, a hash function is to be understood as being a function in the case of which it is not possible to calculate a matching input value for a given function value. In addition, an input character sequence of arbitrary length is allocated an output character sequence of fixed length. Furthermore, additional properties may be demanded for the hash function. One such additional property is freedom from conflict, i.e. it must not be possible to find two different input character sequences which produce the same output character sequence.

The invention is based on the problem of specifying a simplified method for the computer-aided interchange of cryptographic keys which does not presuppose the existence of common secret keys.

This problem is solved by the method according to patent claim 1 and by the arrangement according to patent claim 29.

In the method, a first value is formed from a first random number using a generating element of a finite group in the first computer unit. A first message is transmitted from the first computer unit to the second

computer unit, the first message containing at least the first value. A session key is formed in the second computer unit using a first hash function, a first input variable for the first hash function containing at least one first term which is formed by exponentiation of the first value using a secret network key. The session key is formed in the first computer unit using the first hash function, a second input variable for the first hash function containing at least one second term which is formed by exponentiation of a public network key using the first random number. A fourth input variable is formed in the first computer unit using a second hash function or the first hash function, a third input variable for the first hash function or for the second hash function containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously. A signature term is formed in the first computer unit from at least the fourth input variable using a first signature function. A third message is transmitted from the first computer unit to the second computer unit, the third message containing at least the signature term from the first computer unit. The signature term is verified in the second computer unit.

In the case of the arrangement, the first computer unit and the second computer unit are set up such that the following method steps can be carried out:

- a first value is formed from a first random number using a generating element of a finite group in the first computer unit,
- a first message is transmitted from the first computer unit to the second computer unit, the first message containing at least the first value,
- a session key is formed in the second computer unit using a first hash function, a first input variable for the first hash function containing at least one

first term which is formed by exponentiation of the first value using a secret network key,
- the session key is formed in the first computer unit using the first hash function, a second input variable for the first hash function containing at least one second term which is formed by exponentiation of a public network key using the first random number,
- a fourth input variable is formed in the first computer unit using a second hash function or the first hash function, a third input variable for the first hash function or for the second hash function containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously,
- a signature term is formed in the first computer unit from at least the fourth input variable using a first signature function,
- a third message is transmitted from the first computer unit to the second computer unit, the third message containing at least the signature term from the first computer unit, and
- the signature term is verified in the second computer unit.

The advantages which are achieved by the invention are primarily a considerable reduction in the length of the transmitted messages and the implementation of further security aims.

In addition, the invention can be adapted very easily to different requirements, since there is no restriction to particular algorithms for signature formation and encryption.

Advantageous developments of the invention can be found in the dependent claims.

In one development, provision is made for a long-service secret network key and a long-service public network key to be used.

A long-service key is to be understood below as being a key which is used for a plurality of protocol passes.

The invention and its developments implement the following security aims:

- mutual explicit authentication by the user and the network, i.e. mutual verification of the claimed identity,
- key declaration between the user and the network with mutual implicit authentication, i.e. the method achieves the effect that, after completion of the procedure, a common secret session key is available, of which each party knows that only the authentic counterpart can likewise be in possession of the secret session key,
- assurance for the user that the session key is fresh (up to date),
- mutual confirmation of the session key by the user and the network, i.e. confirmation that the counterpart is actually in possession of the declared secret session key.

The following advantageous developments of the method also relate to these security aims.

In one development, a dependable public user key for the first computer unit, e.g. in the form of a user certificate, is additionally made available in the first computer unit and a dependable public network key for the second computer unit, e.g. in the form of a network certificate, is made available in the second computer unit. The public network key need not be available in the first computer unit in this development.

In a further refinement, it is not necessary for the public user key to be available in the second computer unit.

In accordance with a further refinement, no dependable public network key for the second computer unit is necessary in the first computer unit. A dependable public certification key for the certification computer unit is available in the first computer unit. This means that the first computer unit needs to "acquire" the dependable public network key in the form of a network certificate from a certification computer unit. The second computer unit likewise needs the dependable public user key in the form of a user certificate from the certification computer unit.

The developments of the invention in accordance with patent claims 13, 15 and 20 implement the security aim of user anonymity, i.e. confidentiality of the identity of the user with regard to third parties.

The development of the method according to the invention in accordance with patent claim 15 permits the use of temporary user identities.

The development of the method in accordance with patent claim 16 primarily ensures additional authentication of the second computer unit with regard to the first computer unit.

The development in accordance with patent claim 18 implements the security aim of assurance for the network that the session key is fresh (up to date).

The development in accordance with patent claim 21 additionally implements the security aim of incontestability of data which has been sent from the user to the network.

The drawings show preferred exemplary embodiments of the invention which are described in more detail below.

In the drawings

Figure 1 shows a flowchart illustrating a first exemplary embodiment of the method with a few developments;

Figure 2 shows a flowchart illustrating a second exemplary embodiment of the method with a few developments;

Figure 3 shows a flowchart illustrating a third exemplary embodiment of the method with a few developments.

First exemplary embodiment

Figure 1 shows a sketch of the execution of the method. The method relates to the interchange of cryptographic keys between a first computer unit U and a second computer unit N, where the first computer unit U is to be understood as being a computer unit of a user of a mobile radio network and a second computer unit N is to be understood as being a computer unit of the network operator of a mobile radio system.

It is a prerequisite for the method that a dependable public network key g^s for the second computer unit N is available in the first computer unit U and that a dependable public user key KU for the first computer unit U is available in the second computer unit N, where g is a generating element of a finite group.

In the first computer unit U, a first random number t is generated (step 101). The generating element g of a finite group is used to form a first value g^t from the first random number t in the first computer unit U (step 102).

Once the first value g^t has been calculated, a first message M1, containing at least the first value g^t, is coded. The first message M1 is transmitted from the first computer unit U to the second computer unit N (step 103).

In the second computer unit N, the first message M1 is decoded. The first message M1 may also be transmitted over an insecure channel, that is to say also via an air interface, in unencrypted form, since the logarithmation of the first value g^t cannot be performed in a reasonable time.

In the second computer unit N, a second random number r is generated (step 104). This additional method step implements an additional security aim: the assurance for the second computer unit N that a session key K described below is fresh (up to date).

In the second computer unit N, a first hash function h1 is used to form a session key K (step 105). At least one first term is used as a first input variable for the first hash function h1. The first term is formed by raising the first value g^t to a higher power using a secret network key s.

If the second random number r is used, the first input variable for the first hash function h1 additionally contains at least the second random number r.

A response A is now formed in the second computer unit N (step 106). Various variants are provided for forming the response A. Thus, for example, it is possible for an encryption function Enc to be used to encrypt a constant const, and possibly further variables, with the session key K. The constant const is known both to the first computer unit U and to the second computer unit N. The encryption function Enc is also known both

to the second computer unit N and to the first computer unit U as the encryption function which is to be used in the method.

A further option for forming the response A (step 106) is for the session key K, and possibly prescribable further variables, e.g. an identity statement idN for the second computer unit N and/or the second random number, to be used as input variable for a second hash function h2, and for the "hashed" value for the session key K, and possibly for the further variables, to be used as response A.

Stringing together the second random number r, the response A and also an optional first data field dat1 forms a second message M2. The optional first data field dat1 is only contained in the second message M2 if this is provided in the method.

The second message M2 is coded in the second computer unit N and is transmitted to the first computer unit U (step 107).

In the first computer unit U, the second message M2 is decoded, which means that the first computer unit U has the second random number r, the response A and possibly the optional first data field dat1 available. The length of the optional first data field dat1 may be of any desired size, i.e. it is also possible for the optional first data field dat1 not to be present.

In the first computer unit U, the session key K is now likewise formed (step 108), using the first hash function h1, which is known both to the second computer unit N and to the first computer unit U. A second input variable for the first hash function h1 for forming the session key K in the first computer unit U contains at least one second term. The second term is formed from

exponentiation of a public network key g^s using the first random number t. If the use of the second random number r is provided in the method for calculating the session key K, the second input variable for the first hash function h1 for forming the session key K in the first computer unit U additionally contains the second random number r.

The use of the first random number t and of the second random number r for generating the session key K ensures that the session key K is up to date, since the first random number t and the second random number r are respectively used only for one session key K in each case. This prevents reinjection of an older key as the session key K.

Once the session key K has been formed in the first computer unit U, the received response A is used to check whether the session key K formed in the first computer unit U matches the session key K which was formed in the second computer unit N (step 109). Subject to the variants described above for forming the response A, various options are provided for checking the session key K using the response A.

One option is that, if the response A has been formed in the second computer unit N by encrypting the constant const, and possibly further variables, with the session key K using the encryption function Enc, the response A is decrypted, and hence the first computer unit U receives a decrypted constant const', and possibly prescribable further variables, which is/are compared with the known constant const, and possibly the further variables.

The session key K may also be checked, using the response A, by encrypting the constant const, known to the first computer unit U, and possibly prescribable

further variables, with the session key K, formed in the first computer unit U, using the encryption function Enc and checking the result with the response A for a match. This procedure is also used when the response A is formed in the second computer unit N, by applying the second hash function h2 to the session key K, and possibly to the further variables. In this case, the session key K formed in the first computer unit U, and possibly prescribable further variables, is/are used as input variable for the second hash function h2 in the first computer unit U. The "hashed" value for the session key K formed in the first computer unit U, and possibly for further variables, is then checked with the response A for a match. This achieves the aim of key confirmation for the session key K.

As a result of the secret network key s being used for calculating the session key K in the second computer unit N, and the public network key g^s being used for calculating the session key K in the first computer unit U, the second computer unit N is authenticated by the first computer unit U. This is achieved provided that it is known for the first computer unit U that the public network key g^s actually belongs to the second computer unit N.

Subsequent to confirmation of the session key K by means of a check on the response A, a signature term is calculated (step 110). To this end, a third hash function h3 is used to form a fourth input variable. The third hash function h3 can, but need not, be the same hash function as the first hash function h1 and/or the second hash function h2. As a third input variable for the third hash function h3, a term is used which contains one or more variables from which it is possible to infer the session key unambiguously. In addition, the third input variable may contain the optional first data field dat1 or else an optional

second data field dat2, if the use thereof is provided in the method.

Such variables are the first value g^t, the public network key g^s and the second random number r.

It is subsequently not possible to contest the fact that the data contained in the first optional data field dat1 and in the second optional data field dat2 has been sent from the first computer unit U.

The data contained in the first optional data field dat1 and in the second optional data field dat2 may be telephone numbers, the current time or similar parameters suitable for this purpose. This information may be used as a tool for incontestable charge accounting.

A first signature function Sig_U is used to form the signature term from at least the fourth input variable. To achieve a higher degree of security, the signature term may be encrypted. In this case, the signature term is encrypted with the session key K using the encryption function Enc and forms the first encrypted term VT1.

In addition, if the security aim of "anonymity of the user" is to be implemented, a second encrypted term VT2 is calculated by encrypting an identity variable IMUI for the first computer unit U with the session key K using the encryption function Enc. When an optional second data field dat2 is used, a third encrypted term VT3 is calculated in the first computer unit U by encrypting the optional second data field dat2 with the session key K using the encryption function Enc; the optional second data field dat2 may also be transmitted in unencrypted form.

The three encrypted terms may also be combined to form a fourth encrypted term VT4, in which the interlinkage of signature term, identity variable IMUI and optional second data field dat2 is encrypted with the session key K (step 111).

In the first computer unit U, a third message M3, containing at least the signature term and the identity variable IMUI for the first computer unit U, is formed and coded.

If anonymity of the first computer unit U is to be ensured, the third message M3 contains, instead of the identity variable IMUI for the first computer unit U, at least either the second encrypted term VT2 or the fourth encrypted term VT4, which contains the information about the identity of the first computer unit U in encrypted form, which can be decrypted only by the second computer unit N.

If the use of the optional second data field dat2 is provided, the third message M3 additionally contains at least the third encrypted term VT3 or the fourth encrypted term VT4 or the optional second data field dat2 in plain text.

If the third message M3 contains the first encrypted term VT1, the second encrypted term VT2 or the third encrypted term VT3 or the fourth encrypted term VT4, these are decrypted in the second computer unit N. This is done for the first encrypted term VT1, which may be present, before verification of the signature term.

The third message M3 is transmitted from the first computer unit U to the second computer unit N (step 112).

In addition, authentication of the first computer unit U for the second computer unit N is ensured by the signature term, which contains the random number r, the use of which guarantees that the third message M3 has actually been sent from the first computer unit U at the present time.

In the second computer unit N, the third message M3 is decoded, decrypted, and a user certificate CertU available to the second computer unit N is then used to verify the signature term (step 113).

If temporary user identities are provided for the method, then the method described above is extended by a few method steps.

The second computer unit N must first be informed of which first computer unit U is to be allocated a new temporary identity variable TMUIN by the second computer unit N.

To this end, an old temporary identity variable TMUIO is transmitted from the first computer unit U to the second computer unit N as an additional component of the first message M1.

Once the first message M1 has been received, the second computer unit N thus knows for which first computer unit U the new temporary identity variable TMUIN is intended.

The new temporary identity variable TMUIN for the first computer unit U is then formed in the second computer unit N. This may be performed, for example, by generating a random number or by means of tables in which potential identity variables are stored. The new temporary identity variable TMUIN for the first computer unit U is used to form a fifth encrypted term

VT5 in the second computer unit N by encrypting the new temporary identity variable TMUIN for the first computer unit U with the session key K using the encryption function Enc.

In this case, the second message M2 additionally contains at least the fifth encrypted term VT5. The fifth encrypted term VT5 is then decrypted in the first computer unit U. The new temporary identity variable TMUIN for the first computer unit U is now available in the first computer unit U.

So that the second computer unit N is also assured of the fact that the first computer unit U has received the new temporary identity variable TMUIN correctly, the third input variable for the first hash function h1 or for the third hash function h3 additionally contains at least the new temporary identity variable TMUIN for the first computer unit U.

Since the information for the new temporary identity variable TMUIN is contained in the signature term in this case, the third message M3 no longer contains the identity variable IMUI for the first computer unit U.

It is also possible for the new temporary identity variable TMUIN not to be integrated into the signature term, but rather for the second encrypted term VT2 to be formed by encrypting, instead of the identity variable IMUI for the first computer unit U, the new temporary identity variable TMUIN with the session key K using the encryption function Enc. In this case, the third message M3 additionally contains the second encrypted term VT2.

The hash functions used in the method, the first hash function h1, the second hash function h2 and the third

hash function h3 can be produced by the same hash functions, or else by different hash functions.

Second exemplary embodiment

Figure 2 shows a sketch of the execution of a second exemplary embodiment of the method.

A prerequisite for this exemplary embodiment of the method is that a dependable public user key KU for the first computer unit U in the form of a user certificate CertU is made available in the first computer unit U, and that a dependable public network key g^s for the second computer unit N in the form of a network certificate CertN is made available in the second computer unit N. The public network key g^s need not be available in the first computer unit U. Likewise, it is not necessary for the public user key KU to be available in the second computer unit N.

In the first computer unit U, the first random number t is generated (step 201). The generating element g of a finite group in the first computer unit U is used to form the first value g^t from the first random number t (step 202).

Once the first value g^t has been calculated, a first message M1 is coded, said first message containing at least the first value g^t and an identity statement id_{CA} for a certification computer unit CA which delivers the network certificate CertN which can be verified by the first computer unit U. The first message M1 is transmitted from the first computer unit U to the second computer unit N (step 203).

In the second computer unit N, the first message M1 is decoded.

As described in figure 2, a second random number r is generated in the second computer unit N (step 204). This additional method step implements an additional security aim: the assurance for the second computer unit N that a session key K described below is fresh (up to date).

In the second computer unit N, the first hash function h1 is used to form the session key K (step 205). The first input variable used for the first hash function h1 is a first term. The first term is formed by raising the first value g^t to a higher power using the secret network key s.

When the second random number r is used, the first input variable for the first hash function h1 additionally contains at least the second random number r.

A response A is now formed in the second computer unit N (step 206). To form the response A, the variants described within the context of the first exemplary embodiment are provided.

Stringing together the second random number r, the network certificate CertN, the response A and an optional first data field dat1 forms the second message M2. The optional first data field dat1 is only contained in the second message M2 if this is provided in the method.

The second message M2 is coded in the second computer unit N and is transmitted to the first computer unit U (step 207).

In the first computer unit U, the second message M2 is decoded, which means that the first computer unit U has the second random number r, the response A and possibly

the optional first data field dat1 available. The length of the optional first data field dat1 can be of any desired size, i.e. it is also possible for the optional first data field dat1 not to be present.

Next, the network certificate CertN contained in the second message M2 is verified in the first computer unit. Hence, the public network key g^s is available in the first computer unit U.

In the first computer unit U, the session key K is now likewise formed (step 208), using the first hash function h1, which is known both in the second computer unit N and in the first computer unit U. A second input variable for the first hash function h1 for forming the session key K in the first computer unit U contains at least one second term. The second term is formed from exponentiation of the public network key g^s using the first random number t. If the use of the second random number r is provided in the method for calculating the session key K, the second input variable for the first hash function h1 for forming the session key K in the first computer unit U additionally contains the second random number r.

The use of the first random number t and of the second random number r for generating the session key K ensures that the session key K is up to date, since the first random number t and the second random number r are respectively used only for one session key K in each case. This prevents reinjection of an older key as the session key K.

Once the session key K has been formed in the first computer unit U, the received response A is used to check whether the session key K formed in the first computer unit U matches the session key K which was formed in the second computer unit N (step 209).

Subject to the variants described above for forming the response A, various options are provided for checking the session key K using the response A.

To check the response A, the variants described within the context of the first exemplary embodiment are provided. This achieves the aim of key confirmation for the session key K.

As a result of the secret network key s being used for calculating the session key K in the second computer unit N, and the public network key g^s being used for calculating the session key K in the first computer unit U, the second computer unit N is authenticated by the first computer unit U. This is achieved provided that it is known for the first computer unit U that the public network key g^s actually belongs to the second computer unit N.

Subsequent to confirmation of the session key K by means of a check on the response A, the signature term is calculated (step 210). To this end, the third hash function h3 is used to form a fourth input variable. The third hash function h3 can, but need not, be the same hash function as the first hash function h1 and/or the second hash function h2. As a third input variable for the third hash function h3, a term is used which contains one or more variables from which it is possible to infer the session key unambiguously. In addition, the third input variable may contain the optional first data field dat1 or else an optional second data field dat2, if the use thereof is provided in the method.

Such variables are the first value g^t, the public network key g^s and the second random number r.

It is subsequently not possible to contest the fact that the data contained in the first optional data field dat1 and in the second optional data field dat2 has been sent from the first computer unit U.

The data contained in the first optional data field dat1 and in the second optional data field dat2 may be telephone numbers, the current time or similar parameters suitable for this purpose. This information may be used as a tool for incontestable charge accounting.

A first signature function Sig_U is used to form the signature term from at least the fourth input variable. To achieve a higher degree of security, the signature term may be encrypted. In this case, the signature term is encrypted with the session key K using the encryption function Enc and forms the first encrypted term VT1.

In addition, if the security aim of "anonymity of the user" is to be implemented, a second encrypted term VT2 is calculated by encrypting a user certificate CertU for the first computer unit U with the session key K using the encryption function Enc. When an optional second data field dat2 is used, a third encrypted term VT3 can be calculated in the first computer unit U by encrypting the optional second data field dat2 with the session key K using the encryption function Enc. The optional second data field dat2 may likewise be transmitted in unencrypted form.

The three encrypted terms may also be combined to form a fourth encrypted term VT4, in which the chain comprising signature term, identity variable IMUI and optional second data field dat2 is encrypted with K (step 211).

In the first computer unit U, a third message M3, containing at least the signature term and the user certificate CertU for the first computer unit U, is formed and coded. If user anonymity of the first computer unit U is to be ensured, the third message M3 contains, instead of the user certificate CertU for the first computer unit U, at least either the second encrypted term VT2 or the fourth encrypted term VT4, which contains the user certificate CertU for the first computer unit U in encrypted form, which can be decrypted only by the second computer unit N.

If the use of the optional second data field dat2 is provided, the third message M3 additionally contains at least the third encrypted term VT3 or the fourth encrypted term VT4. If the third message M3 contains the first encrypted term VT1, the second encrypted term VT2 or the third encrypted term VT3 or the fourth encrypted term VT4, these are decrypted in the second computer unit N. This is done for the first encrypted term VT1, which may be present, before verification of the signature term.

The third message M3 is transmitted from the first computer unit U to the second computer unit N (step 212).

In the second computer unit N, the third message M3 is decoded, decrypted, and a user certificate CertU available to the second computer unit N is then used to verify the signature term (step 213).

In addition, authentication of the first computer unit U for the second computer unit N is ensured by the signature term, which contains the random number r, the use of which guarantees that the third message M3 has actually been sent from the first computer unit U at the present time.

If temporary user identities are provided for the method, then the method described above is extended by a few method steps.

In the second computer unit N, a new temporary identity variable TMUIN is formed for the first computer unit U and is subsequently allocated to the first computer unit U. This may be performed by generating a random number or by means of tables in which potential identity variables are stored. The new temporary identity variable TMUIN for the first computer unit U is used to form a fifth encrypted term VT5 in the second computer unit N by encrypting the new temporary identity variable TMUIN for the first computer unit U with the session key K using the encryption function Enc.

In this case, the second message M2 additionally contains at least the fifth encrypted term VT5. The fifth encrypted term VT5 is then decrypted in the first computer unit U. The new temporary identity variable TMUIN for the first computer unit U is now available in the first computer unit U.

So that the second computer unit N is also assured of the fact that the first computer unit U has received the new temporary identity variable TMUIN correctly, the third input variable for the first hash function h1 or for the third hash function h3 additionally contains at least the new temporary identity variable TMUIN for the first computer unit U.

It is also possible for the new temporary identity variable TMUIN not to be integrated into the signature term, but rather for the second encrypted term VT2 to be formed by encrypting the new temporary identity variable TMUIN for the first computer unit U with the session key K using the encryption function Enc. In

this case, the third message M3 additionally contains the second encrypted term VT2.

Third exemplary embodiment

Figure 3 shows a sketch of the execution of a third exemplary embodiment.

A prerequisite for this exemplary embodiment of the method is that no dependable public network key g^s for the second computer unit N is available in the first computer unit U. A dependable public certification key cs for a certification computer unit CA is available in the first computer unit U. This means that the first computer unit U needs to "acquire" the dependable public network key g^s in the form of a network certificate CertN from the certification computer unit CA. Likewise, the second computer unit N needs the dependable public user key KU in the form of a user certificate CertU from the certification computer unit CA.

In the first computer unit U, the first random number t is generated (step 301). The generating element g of a finite group in the first computer unit U is used to form the first value g^t from the first random number t (step 302).

Once the first value g^t has been calculated, a first message M1 is coded, said first message containing at least the first value g^t, an identity variable IMUI for the first computer unit U and an identity statement id_{CA} for a certification computer unit CA which delivers a network certificate CertN which can be verified by the first computer unit U. This is necessary when a plurality of certification authorities with different secret certification keys are provided. If the security aim of user anonymity is to be implemented, an

intermediate key L is formed in the first computer unit U before formation of the first message M1. This is done by raising the public key declaration key g^u for the certification computer unit CA, which key is available in the first computer unit U, to a higher power using the first random number t. Subsequently, the identity variable IMUI for the first computer unit U is in this case encrypted with the intermediate key L using an encryption function Enc, and the result represents a fifth encrypted term VT5. The fifth encrypted term VT5 is integrated into the first message M1 instead of the identity variable IMUI for the first computer unit U. The first message M1 is transmitted from the first computer unit U to the second computer unit N (step 303).

In the second computer unit N, the first message M1 is decoded and a fourth message M4 is formed (step 304), said fourth message containing a chain comprising the certificate CertN, known to the second computer unit N, for the public network key g^s, the first value g^t and the identity variable IMUI for the first computer unit U. If the security aim of user anonymity is to be implemented, the fifth encrypted term VT5 is coded in the fourth message M4 instead of the identity variable IMUI for the first computer unit U.

The fourth message M4 is coded in the second computer unit N and is then transmitted to the certification computer unit CA (step 304).

The fourth message M4 is decoded in the certification computer unit CA.

Next, if user anonymity is ensured, that is to say the fifth encrypted term VT5 has also been sent in the fourth message M4, the intermediate key L is calculated in the certification computer unit CA by raising the

first value g^t to a higher power using a secret key declaration key u for the certification computer unit CA.

The fifth encrypted term VT5 is decrypted with the intermediate key L using the encryption function Enc, as a result of which the identity variable IMUI for the first computer unit U is known in the certification computer unit CA.

In the certification computer unit CA, the user certificate CertU is then ascertained. The user certificate CertU is ascertained from a dedicated database for the certification computer unit CA, said database containing all the certificates for the computer units for which the certification computer unit CA produces certificates.

To check the validity of the network certificate CertN and of the user certificate CertU, an identity statement id_N for the network computer unit N and the public network key g^s also sent in the fourth message, the identity variable IMUI for the first computer unit U and also the ascertained user certificate CertU are compared with a revocation list containing invalid certificates, keys or identity variables.

The certification computer unit CA then forms three chains of certificates, a first certificate chain CertChain (U, N), a second certificate chain CertChain (N, U) and a third certificate chain CertChain (N, CA).

The first certificate chain CertChain (U, N) can be verified by the first computer unit U using the public certification key for the certification computer unit CA, which is known to the first computer unit U, and contains as last element a certificate CertN for the public key g^s from the second computer unit N.

The second certificate chain CertChain (N, U) can be verified by the second computer unit N and contains as last element a certificate CertU for the public key KU from the first computer unit U.

The third certificate chain CertChain (N, CA) can be verified by the second computer unit N and contains as last element a certificate for the public verification key from the certification computer unit CA.

The first certificate chain CertChain (U, N) and the second certificate chain CertChain (N, U) can be uniquely identified by the identifiers cidU and cidN.

Next, a third term is formed from at least one chain comprising the first value g^t and the identifiers cidU and cidN.

The third term is "hashed" using a fourth hash function h4, and the result of the hash function h4 is signed using a third signature function Sig_{CA}.

In addition, a time stamp TS is created in the certification computer unit CA. This time stamp is optionally included in the third term.

A fifth message M5 formed in the certification computer unit CA contains at least one chain comprising the signed third term and the certificate chains CertChain (U, N) and CertChain (N, U), and also optionally the time stamp TS and the certificate chain CertChain (N, CA). The signed hash value for the third term and also the certificate chain CertChain (N, U) are optionally encrypted using the intermediate key L.

The fifth message M5 is coded in the certification computer unit CA and is transmitted to the second

computer unit N (step 305). Once the fifth message M5 is decoded in the second computer unit N, the signed hash value for the third term is verified, provided that it is not encrypted with L.

In the second computer unit N, a fourth term is now formed, said fourth term containing at least one chain comprising the certificate chain CertChain (U, N) and the signed hash value (optionally encrypted with the intermediate key L) for the third term.

In the second computer unit N, the first hash function h1 is used to form a session key K. A first input variable used for the first hash function h1 is a concatenation of a first term with the second random number r. The first term is formed by raising the first value g^t to a higher power using a secret network key s. The second random number r is used when the intention is to implement the additional security aim of assurance for the second computer unit N that the session key K is fresh (up to date). If this security aim is not required, the second random number r is not used in the method for calculating the session key K.

In the second computer unit N, a response A is formed. For forming the response A, the variants described in the first exemplary embodiment are provided.

Stringing together the second random number r, the fourth term, the response A and also an optional first data field dat1 and the optional time stamp forms a second message M2. The optional first data field dat1 is only contained in the second message M2 if this is provided in the method.

The second message M2 is coded in the second computer unit N and is transmitted to the first computer unit U (step 306).

In the first computer unit U, the second message M2 is decoded, which means that the first computer unit U has the second random number r, the response A and also possibly the optional first data field dat1 and possibly the time stamp TS available. The length of the optional first data field dat1 can be of any desired size, i.e. it is also possible for the optional first data field dat1 not to be present.

In the first computer unit U, the session key K is now likewise formed (step 307), using the first hash function h1, which is known both to the second computer unit N and to the first computer unit U. A second input variable for the first hash function h1 for forming the session key K in the first computer unit U contains at least one second term. The second term is formed from exponentiation of a public network key g^s using the first random number t. If the second random number r is provided in the method for calculating the session key K, the second input variable for the first hash function h1 for forming the session key K in the first computer unit U additionally contains the second random number r.

Once the session key K has been formed in the first computer unit U, the received response A is used to check whether the session key K formed in the first computer unit U matches the session key K which was formed in the second computer unit N (step 308).

Subject to the variants described above for forming the response A, the options described above are provided for checking the session key K using the response A.

As a result of the secret network key s being used for calculating the session key K in the second computer unit N, and the public network key g^s being used for calculating the session key K in the first computer

unit U, the second computer unit N is authenticated by the first computer unit U. This is achieved provided that it is known for the first computer unit U that the public network key g^s actually belongs to the second computer unit N. That is achieved by U as a result of verification of the certificate chain CertChain (U, N) and also of the signed hash value for the third term. If the latter is encrypted with the intermediate key L, it needs to be decrypted using the intermediate key L before verification.

Subsequent to confirmation of the session key K by means of a check on the response A, a signature term is calculated (step 309). To this end, a third hash function h3 is used to form a fourth input variable. The third hash function h3 can, but need not, be the same hash function as the first hash function h1 and/or the second hash function h2. As a third input variable for the third hash function h3, a term is used which contains one or more variables from which it is possible to infer the session key unambiguously. In addition, the third input variable may contain the optional first data field dat1 or else an optional second data field dat2, if the use thereof is provided in the method.

Such variables are the first value g^t, the public network key g^s and the second random number r.

It is subsequently not possible to contest the fact that the data contained in the first optional data field dat1 and in the second optional data field dat2 is sent from the first computer unit U.

The data contained in the first optional data field dat1 and in the second optional data field dat2 may be telephone numbers, the current time or similar parameters suitable for this purpose. This information

may be used as a tool for incontestable charge accounting.

A first signature function Sig_U is used to form the signature term from at least the fourth input variable. To achieve a higher degree of security, the signature term may be encrypted. In this case, the signature term is encrypted with the session key K using the encryption function Enc and forms the first encrypted term VT1.

When an optional second data field dat2 is used, a third encrypted term VT3 is calculated in the first computer unit U by encrypting the optional second data field dat2 with the session key K using the encryption function Enc. The optional second data field dat2 may also be transmitted in unencrypted form, that is to say in plain text.

As an alternative for forming the first and the third encrypted term VT1 and VT3, it is also possible for a fourth encrypted term VT4 to be formed by encrypting at least the chain comprising the signature term and optionally the data field dat2 and the intermediate key L using the session key K (step 310).
In the first computer unit U, a third message M3 is formed and coded, said third message comprising at least the first encrypted term VT1 and, if the optional second data field dat2 is used, the third encrypted term VT3 or the optional second data field dat2 in plain text, or else comprising the fourth encrypted term VT4.

The third message M3 is transmitted from the first computer unit U to the second computer unit N (step 311).

In the second computer unit N, the third message M3 is decoded and then the first encrypted term VT1 and also possibly the third encrypted term VT3, or else the fourth encrypted term VT4, is decrypted. If parts of the message M5 have been encrypted with L, then the second computer unit N can now use the intermediate key L received in message M3 to decrypt the encrypted parts of the message M5. The second computer unit N can then verify the second certificate chain Cert (N, U) and also the signed hash value for the third term using the public verification key of CA. The user certificate CertU, which is now available to the second computer unit N, is used to verify the signature term.

In addition, authentication of the first computer unit U for the second computer unit N is ensured by the signature term in the third message M3, which contains the random number r, the use of which also guarantees that the third message M3 has actually been sent from the first computer unit U at the present time.

If temporary user identities are provided for the method, then the method described above is extended by a few method steps.

In the second computer unit N, a new temporary identity variable TMUIN is formed for the first computer unit U and is subsequently allocated to the first computer unit U. This may be performed, for example, by generating a random number or by means of tables in which potential identity variables are stored. The new temporary identity variable TMUIN for the first computer unit U is used to form a fifth encrypted term VT5 in the second computer unit N by encrypting the new temporary identity variable TMUIN for the first computer unit U with the session key K using the encryption function Enc.

In this case, the second message M2 additionally contains at least the fifth encrypted term VT5. The fifth encrypted term VT5 is then decrypted in the first computer unit U. The new temporary identity variable TMUIN for the first computer unit U is now available in the first computer unit U.

So that the second computer unit N is also assured of the fact that the first computer unit U has received the new temporary identity variable TMUIN correctly, the third input variable for the first hash function h1 or for the second hash function h2 additionally contains at least the new temporary identity variable TMUIN for the first computer unit U.

A few alternatives to the exemplary embodiments described above are illustrated below:

The invention is not restricted to a mobile radio system, and hence it is also not restricted to a user of a mobile radio system and to the network, but rather may be used in all areas in which cryptographic key interchange between two communication parties is required. This may be the case, for example, in a communication link between two computers wishing to interchange data in encrypted form. Without any restriction to the general validity, a first communication party was called the first computer unit U and a second communication party was called the second computer unit N above.

The following publications were cited as part of this document:

[1] A. Aziz, W. Diffie, "Privacy and Authentication for Wireless Local Area Networks", IEEE Personal Communications, 1994, pp. 25 to 31

[2] M. Beller, "Proposed Authentication and Key Agreement Protocol for PCS", Joint Experts Meeting on Privacy and Authentication for Personal Communications, P&A JEM 1993, 1993, pp. 1 to 11

[3] C. Carroll, Y. Frankel, Y. Tsiounis, "Efficient key distribution for slow computing devices", Conference Security&Privacy, Oakland, 1998

[4] J. Zhou, K. Lam, "Undeniable billing in mobile communications", preprint 1998

[5] US 5 214 700

[6] DE brochures: Telesec. Telekom, Produktentwicklung Telesec beim Fernmeldeamt Siegen [Telesec. Telecom, product development Telesec at the Siegen exchange], pp. 12-13

[7] US 5 222 140

[8] US 5 153 919

[9] M. Beller et al, Privacy and Authentication on a Portable Communication System, IEEE Journal on Selected Areas in Communications, Vol. 11, No. 6, pp. 821-829, 1993

[10] DE 195 18 5465 C1

[11] W. Diffie, P. C. van Oorschot, M. Wiener, "Authentication and authenticated key exchanges", Designs, Codes and Cryptography, Vol. 2, pp. 107-125, 1992

Patent claims

1. A method for the computer-aided interchange of cryptographic keys between a first computer unit (U) and a second computer unit (N),
   - in which a first value (g^t) is formed from a first random number (t) using a generating element (g) of a finite group in the first computer unit (U),
   - in which a first message (M1) is transmitted from the first computer unit (U) to the second computer unit (N), the first message (M1) containing at least the first value (g^t),
   - in which a session key (K) is formed in the second computer unit (N) using a first hash function (h1), a first input variable for the first hash function (h1) containing at least one first term which is formed by exponentiation of the first value (g^t) using a secret network key (s),
   - in which the session key (K) is formed in the first computer unit (U) using the first hash function (h1), a second input variable for the first hash function (h1) containing at least one second term which is formed by exponentiation of a public network key (g^s) using the first random number (t),
   - in which a fourth input variable is formed in the first computer unit (U) using a second hash function (h2) or the first hash function (h1), a third input variable for the first hash function (h1) or for the second hash function (h2) containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously,

- in which a signature term is formed in the first computer unit (U) from at least the fourth input variable using a first signature function (Sig_U),
- in which a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing at least the signature term from the first computer unit (U), and
- in which the signature term is verified in the second computer unit (N).

2. The method as claimed in claim 1, in which the secret network key and/or the public network key is/are long-service keys.

3. The method as claimed in claim 1 or 2, in which the third input variable contains a plurality of variables which can be used to infer the session key unambiguously.

4. The method as claimed in one of claims 1 to 3, in which the variable or the variables contains or contain at least the first value (g^t) and/or the public network key (g^s).

5. The method as claimed in one of claims 1 to 4,
- in which the first message (M1) contains an identity statement (id_{CA}) for a certification computer unit (CA) which delivers a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the network certificate

(CertN) or the chain of certificates, the last of which is the network certificate (CertN), and

- in which the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), is verified in the first computer unit (U).

6. The method as claimed in claim 5,
- in which a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing a user certificate (CertU) or a chain of certificates, the last of which is the user certificate (CertU),
- in which the user certificate (CertU) or the chain of certificates, the last of which is the user certificate (CertU), is verified in the second computer unit (N).

7. The method as claimed in one of claims 1 to 6,
- in which the first message(M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement (id_{CA}) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) which can be verified by the first computer unit (U),
- in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the first value (g^t) as input variable,
- in which a fifth message (M5), containing at least the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate

(CertU), is transmitted from the certification computer unit (CA) to the second computer unit (N).

8. The method as claimed in one of claims 1 to 7,
- in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the public network key (g^s), the first value (g^t), the identity variable (IMUI) for the first computer unit (U) as input variable, and an output variable from a third hash function (h3) being signed using a second signature function (Sig_N),
- in which the first signed term is verified in the certification computer unit (CA),
- in which a third term, containing at least the first value (g^t), the public network key (g^s) and an identity statement (id_N) for the second computer unit (N), is formed in the certification computer unit (CA),
- in which a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- in which the hash value for the third term is signed in the certification computer unit (CA) using a third signature function (Sig_{CA}),
- in which a network certificate (CertN) containing at least the third term and the signed hash value for the third term is formed in the certification computer unit (CA),
- in which a fourth hash function (h4) is applied in the certification computer unit (CA) to a fifth term, containing at least the identity statement (id_N) for the second computer unit (N) and a user certificate (CertU),
- in which the hash value for the fifth term is signed using the secret certification key (cs)

by using the third signature function (Sig_{CA}), and the result represents the second signed term,

- in which a fifth message (M5), containing at least the network certificate (CertN), the fifth term and the second signed term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- in which the network certificate (CertN) and the second signed term are verified in the second computer unit (N),
- in which a fourth term, containing at least the public network key (g^s) and the signed hash value for the third term, is formed in the second computer unit (N),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the fourth term, and
- in which the network certificate (CertN) is verified in the first computer unit (U).

9. The method as claimed in one of claims 1 to 8,
   - in which the first message (M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement (id_{CA}) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
   - in which a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least one certificate for the public network key (g^s), the first value

(g^t) and the identity variable (IMUI) for the first computer unit (U),

- in which a third term, containing at least the public network key (g^s) or a variable which determines the public network key (g^s) unambiguously, is formed in the certification computer unit (CA),
- in which a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- in which the hash value for the third term is signed in the certification computer unit (CA) using a third signature function (Sig_{CA}),
- in which a fifth message (M5), containing at least the signed hash value for the third term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- in which the signed hash value for the third term is verified in the second computer unit (N),
- in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the signed hash value for the third term, and
- in which the signed hash value for the third term is verified in the first computer unit (U).

10. The method as claimed in claim 9, in which the third term contains a public user signature key (KU) or a variable which determines the user signature key (KU) unambiguously.

11. The method as claimed in claim 9 or 10, in which the fifth message (M5) and the second message (M2) have at least one chain of certificates.

12. The method as claimed in claim 8, in which the fifth term has a time stamp (TS).

13. The method as claimed in one of claims 9 to 12, in which the third term has a time stamp (TS).

14. The method as claimed in one of claims 7 to 13,
    - in which an intermediate key (L) is formed in the first computer unit (U), before formation of the first message (M1), by raising a public key declaration key (g^u) to a higher power using the first random number (t),
    - in which a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the first message (M1), from the identity variable (IMUI) for the first computer unit (U) by encrypting the identity variable (IMUI) with the intermediate key (L) using an encryption function (Enc),
    - in which the first message (M1) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U),
    - in which the fourth message (M4) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U).

15. The method as claimed in one of claims 7 to 14, in which the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is encrypted with L in the fifth message (M5).

16. The method as claimed in one of claims 7 to 15,

in which at least one of the variables, the identity statement (id_N) for the second computer unit (N), the identity variable (IMUI) for the first computer unit (U), the public network key (g^s), the network certificate (CertN) or the user certificate (CertU) is checked in the certification computer unit (CA) using a revocation list.

17. The method as claimed in one of claims 1 to 16,
- in which the first message (M1) contains at least one old temporary identity variable (TMUIO) for the first computer unit (U),
- in which a new temporary identity variable (TMUIN) is formed for the first computer unit (U) in the second computer unit (N) after the first message (M1) has been received and before the second message (M2) is formed,
- in which a fifth encrypted term (VT5) is formed from the new temporary identity variable (TMUIN) for the first computer unit (U) by encrypting the new temporary identity variable (TMUIN) for the first computer unit (U) with the session key (K) using the encryption function (Enc),
- in which the second message (M2) contains at least the fifth encrypted term (VT5),
- in which the fifth encrypted term (VT5) is decrypted in the first computer unit (U) after the second message (M2) has been received and before the fourth input variable is formed,
- in which the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the new temporary identity variable (TMUIN) for the first computer unit (U) for the purpose of forming the fourth input variable, and

- in which the third message (M3) does not contain the identity variable (IMUI) for the first computer unit (U).

18. The method as claimed in one of claims 1 to 17,
    - in which a response (A) containing information about the session key (K) is formed in the second computer unit (N),
    - in which a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the response (A), and
    - in which the session key (K) is checked in the first computer unit (U) using the response (A).

19. The method as claimed in one of claims 1 to 18, in which the third message (M3) contains an identity variable (IMUI) for the first computer unit (U).

20. The method as claimed in one of claims 1 to 19,
    - in which the first input variable for the first hash function (h1) contains at least one second random number (r) in the second computer unit (N),
    - in which the second message (M2) contains the second random number (r), and
    - in which the second input variable for the first hash function (h1) contains at least the second random number (r) in the first computer unit (U).

21. The method as claimed in one of claims 1 to 20, in which the variable or the variables as claimed in claim 3 contains or contain the second random number (r).

22. The method as claimed in one of claims 1 to 21,

- in which a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the third message (M3), from the identity variable (IMUI) for the first computer unit (U) by encrypting at least the identity variable (IMUI) with the session key (K) using the encryption function (Enc),
- in which the third message (M3) contains the second encrypted term (VT2), and
- in which the second encrypted term (VT2) is decrypted in the second computer unit (N) after the third message (M3) has been received.

23. The method as claimed in one of claims 1 to 22,
- in which the second message (M2) contains an optional first data field (dat1), and
- in which the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the optional first data field (dat1) for the purpose of forming the fourth input variable.

24. The method as claimed in one of claims 1 to 23,
- in which a third encrypted term (VT3) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least one optional second data field (dat2) with the session key (K) using the encryption function (Enc),
- in which the third message (M3) contains at least the third encrypted term (VT3), and
- in which the third encrypted term (VT3) is decrypted in the second computer unit (N) after the third message (M3) has been received.

25. The method as claimed in one of claims 1 to 24,
- in which a first encrypted term (VT1) is formed in the first computer unit (U), before formation

of the third message (M3), by encrypting at least the signature term using the encryption function (Enc),
- in which the third message (M3) contains the first encrypted term (VT1), and
- in which the first encrypted term (VT1) is decrypted in the second computer unit (N) after the third message (M3) has been received and before the signal term is verified.

26. The method as claimed in one of claims 1 to 25, in which a response (A) is formed in the second computer unit (N) by encrypting a constant (const), and possibly further variables, which are known in the second computer unit (N) and in the first computer unit (U), with the session key (K) using the encryption function (Enc).

27. The method as claimed in one of claims 1 to 26, in which the response (A) is checked in the first computer unit (U) by encrypting a constant (const), and possibly further variables, with the session key (K) using the encryption function (Enc) and comparing the result with the response (A).

28. The method as claimed in one of claims 1 to 26, in which the response (A) is checked in the first computer unit (U) by decrypting the response (A) with the session key (K) using the encryption function (Enc) and comparing a decrypted constant (const') with a constant (const), and possibly further variables.

29. The method as claimed in one of claims 1 to 28,
- in which a response (A) is formed in the second computer unit (N) by applying a third hash

function (h3) to an input variable which contains at least the session key (K), and
- in which the response (A) is checked in the first computer unit (U) by applying the third hash function (h3) to the input variable, which contains at least the session key (K), and comparing the result with the response (A).

30. The method as claimed in one of claims 1 to 29, in which the third message (M3) contains at least one optional second data field (dat2).

31. The method as claimed in one of claims 1 to 30, in which the first computer unit (U) is formed by a mobile communication terminal and/or the second computer unit (N) is formed by an authentication unit in a mobile communication network.

32. An arrangement for the computer-aided interchange of cryptographic keys between a first computer unit (U) and a second computer unit (N), in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a first value (g^t) is formed from a first random number (t) using a generating element (g) of a finite group in the first computer unit (U),
- a first message (M1) is transmitted from the first computer unit (U) to the second computer unit (N), the first message (M1) containing at least the first value (g^t),
- a session key (K) is formed in the second computer unit (N) using a first hash function (h1), a first input variable for the first hash function (h1) containing at least one first term which is formed by exponentiation of the first value (g^t) using a secret network key (s),

- the session key (K) is formed in the first computer unit (U) using the first hash function (h1), a second input variable for the first hash function (h1) containing at least one second term which is formed by exponentiation of a public network key (g^s) using the first random number (t),
- a fourth input variable is formed in the first computer unit (U) using a second hash function (h2) or the first hash function (h1), a third input variable for the first hash function (h1) or for the second hash function (h2) containing, for the purpose of forming the fourth input variable, one or more variables which can be used to infer the session key unambiguously,
- a signature term is formed in the first computer unit (U) from at least the fourth input variable using a first signature function (Sig_U),
- a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing at least the signature term from the first computer unit (U), and
- the signature term is verified in the second computer unit (N).

33. The arrangement as claimed in claim 31, in which the secret network key and/or the public network key is/are long-service keys.

34. The arrangement as claimed in claim 32 or 33, in which the first computer unit (U) and the second computer unit (N) are set up such that the third input variable contains a plurality of variables which can be used to infer the session key unambiguously.

35. The arrangement as claimed in one of claims 32 to 34, in which the first computer unit (U) and the second computer unit (N) are set up such that the variable or the variables contains or contain at least the first value (g^t) and/or the public network key (g^s).

36. The arrangement as claimed in one of claims 32 to 35, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first message (M1) contains an identity statement (id_{CA}) for a certification computer unit (CA) which delivers a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), and
- the network certificate (CertN) or the chain of certificates, the last of which is the network certificate (CertN), is verified in the first computer unit (U).

37. The arrangement as claimed in claim 36, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a third message (M3) is transmitted from the first computer unit (U) to the second computer unit (N), the third message (M3) containing a user certificate (CertU) or a chain of

certificates, the last of which is the user certificate (CertU),

- the user certificate (CertU) or the chain of certificates, the last of which is the user certificate (CertU), is verified in the second computer unit (N).

38. The arrangement as claimed in one of claims 32 to 37,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first message(M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement (id_{CA}) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) which can be verified by the first computer unit (U),
- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the first value (g^t) as input variable,
- a fifth message (M5), containing at least the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is transmitted from the certification computer unit (CA) to the second computer unit (N).

39. The arrangement as claimed in one of claims 32 to 38,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least the public network key (g^s), the first value (g^t), the identity variable (IMUI) for the first computer unit (U) as input variable, and an output variable from a third hash function (h3) being signed using a second signature function (Sig_N),
- the first signed term is verified in the certification computer unit (CA),
- a third term, containing at least the first value (g^t), the public network key (g^s) and an identity statement (id_N) for the second computer unit (N), is formed in the certification computer unit (CA),
- a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- the hash value for the third term is signed in the certification computer unit (CA) using a third signature function (Sig_{CA}),
- a network certificate (CertN) containing at least the third term and the signed hash value for the third term is formed in the certification computer unit (CA),
- a fourth hash function (h4) is applied in the certification computer unit (CA) to a fifth term, containing at least the identity statement (id_N) for the second computer unit (N) and a user certificate (CertU),
- the hash value for the fifth term is signed using the secret certification key (cs) by using the third signature function (Sig_{CA}), and the result represents the second signed term,
- a fifth message (M5), containing at least the network certificate (CertN), the fifth term and the second signed term, is transmitted from the

certification computer unit (CA) to the second computer unit (N),

- the network certificate (CertN) and the second signed term are verified in the second computer unit (N),
- a fourth term, containing at least the public network key (g^s) and the signed hash value for the third term, is formed in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the fourth term, and
- the network certificate (CertN) is verified in the first computer unit (U).

40. The arrangement as claimed in one of claims 33 to 39,

in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

- the first message (M1) contains an identity variable (IMUI) for the first computer unit (U) and an identity statement (id_{CA}) for a certification computer unit (CA) which delivers to the first computer unit (U) a network certificate (CertN) or a chain of certificates, the last of which is the network certificate (CertN), which can be verified by the first computer unit (U),
- a fourth message (M4) is transmitted from the second computer unit (N) to the certification computer unit (CA), the fourth message (M4) containing at least one certificate for the public network key (g^s), the first value (g^t) and the identity variable (IMUI) for the first computer unit (U),

- a third term, containing at least one public network key (g^s) or a variable which determines the public network key (g^s) unambiguously, is formed in the certification computer unit (CA),
- a hash value for the third term is formed in the certification computer unit (CA) using a fourth hash function (h4),
- the hash value for the third term is signed in the certification computer unit (CA) using a third signature function (Sig_{CA}),
- a fifth message (M5), containing at least the signed hash value for the third term, is transmitted from the certification computer unit (CA) to the second computer unit (N),
- the signed hash value for the third term is verified in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the signed hash value for the third term, and
- the signed hash value for the third term is verified in the first computer unit (U).

41. The arrangement as claimed in claim 40, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out: the third term contains the public user signature key (KU) or a variable which determines the user signature key (KU) unambiguously.

42. The arrangement as claimed in claim 40 or 41, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out: the fifth message (M5) and the second message (M2) contain at least one chain of certificates.

43. The arrangement as claimed in one of claims 38 to 42, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out: the fifth term has a time stamp (TS).

44. The arrangement as claimed in one of claims 38 to 43, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out: the third term has a time stamp (TS).

45. The arrangement as claimed in claims 38 to 44, in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- an intermediate key (L) is formed in the first computer unit (U), before formation of the first message (M1), by raising a public key declaration key (g^u) to a higher power using the first random number (t),
- a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the first message (M1), from the identity variable (IMUI) for the first computer unit (U) by encrypting the identity variable (IMUI) with the intermediate key (L) using an encryption function (Enc),
- the first message (M1) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U),
- the fourth message (M4) contains the second encrypted term (VT2) instead of the identity variable (IMUI) for the first computer unit (U).

46. The arrangement as claimed in one of claims 38 to 45,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the network certificate (CertN) or a certificate chain, the last element of which is the network certificate (CertN), or the user certificate (CertU) or a certificate chain, the last element of which is the user certificate (CertU), is encrypted with L in the fifth message (M5).

47. The arrangement as claimed in one of claims 38 to 46,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
at least one of the variables, the identity statement (id_N) for the second computer unit (N), the identity variable (IMUI) for the first computer unit (U), the public network key (g^s), the network certificate (CertN) or the user certificate (CertU) is checked in the certification computer unit (CA) using a revocation list.

48. The arrangement as claimed in one of claims 32 to 47,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first message (M1) contains at least one old temporary identity variable (TMUIO) for the first computer unit (U),
- a new temporary identity variable (TMUIN) is formed for the first computer unit (U) in the second computer unit (N) after the first message

(M1) has been received and before the second message (M2) is formed,
- a fifth encrypted term (VT5) is formed from the new temporary identity variable (TMUIN) for the first computer unit (U) by encrypting the new temporary identity variable (TMUIN) for the first computer unit (U) with the session key (K) using the encryption function (Enc),
- the second message (M2) contains at least the fifth encrypted term (VT5),
- the fifth encrypted term (VT5) is decrypted in the first computer unit (U) after the second message (M2) has been received and before the fourth input variable is formed,
- the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the new temporary identity variable (TMUIN) for the first computer unit (U) for the purpose of forming the fourth input variable, and
- the third message (M3) does not contain the identity variable (IMUI) for the first computer unit (U).

49. The arrangement as claimed in one of claims 32 to 48,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a response (A) containing information about the session key (K) is formed in the second computer unit (N),
- a second message (M2) is transmitted from the second computer unit (N) to the first computer unit (U), the second message (M2) containing at least the response (A), and
- the session key (K) is checked in the first computer unit (U) using the response (A).

50. The arrangement as claimed in one of claims 32 to 49,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the third message (M3) contains an identity variable (IMUI) for the first computer unit (U).

51. The arrangement as claimed in one of claims 32 to 48,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the first input variable for the first hash function (h1) contains at least one second random number (r) in the second computer unit (N),
- the second message (M2) contains the second random number (r), and
- the second input variable for the first hash function (h1) contains at least the second random number (r) in the first computer unit (U).

52. The arrangement as claimed in one of claims 32 to 47,
in which the first computer unit (U) and the second computer unit (N) are set up such that the variable or the variables as claimed in claim 34 contains or contain the second random number (r).

53. The arrangement as claimed in one of claims 32 to 51,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a second encrypted term (VT2) is formed in the first computer unit (U), before formation of the

third message (M3), from the identity variable (IMUI) for the first computer unit (U) by encrypting at least the identity variable (IMUI) with the session key (K) using the encryption function (Enc),

- the third message (M3) contains the second encrypted term (VT2), and
- the second encrypted term (VT2) is decrypted in the second computer unit (N) after the third message (M3) has been received.

54. The arrangement as claimed in one of claims 32 to 53,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- the second message (M2) contains an optional first data field (dat1), and
- the third input variable for the first hash function (h1) or for the second hash function (h2) contains at least the optional first data field (dat1) for the purpose of forming the fourth input variable.

55. The arrangement as claimed in one of claims 32 to 54,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a third encrypted term (VT3) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least one optional second data field (dat2) with the session key (K) using the encryption function (Enc),
- the third message (M3) contains at least the third encrypted term (VT3), and

- the third encrypted term (VT3) is decrypted in the second computer unit (N) after the third message (M3) has been received.

56. The arrangement as claimed in one of claims 32 to 55,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a first encrypted term (VT1) is formed in the first computer unit (U), before formation of the third message (M3), by encrypting at least the signature term using the encryption function (Enc),
- the third message (M3) contains the first encrypted term (VT1), and
- the first encrypted term (VT1) is decrypted in the second computer unit (N) after the third message (M3) has been received and before the signal term is verified.

57. The arrangement as claimed in one of claims 32 to 56,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
a response (A) is formed in the second computer unit (N) by encrypting a constant (const), and possibly further variables, which are known in the second computer unit (N) and in the first computer unit (U), with the session key (K) using the encryption function (Enc).

58. The arrangement as claimed in one of claims 44 to 57,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

the response (A) is checked in the first computer unit (U) by encrypting a constant (const), and possibly further variables, with the session key (K) using the encryption function (Enc) and comparing the result with the response (A).

59. The arrangement as claimed in one of claims 44 to 57,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the response (A) is checked in the first computer unit (U) by decrypting the response (A) with the session key (K) using the encryption function (Enc) and comparing a decrypted constant (const'), and possibly further variables, with a constant (const).

60. The arrangement as claimed in one of claims 32 to 59,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
- a response (A) is formed in the second computer unit (N) by applying a third hash function (h3) to an input variable which contains at least the session key (K), and
- the response (A) is checked in the first computer unit (U) by applying the third hash function (h3) to an input variable, which contains at least the session key (K), and comparing the result with the response (A).

61. The arrangement as claimed in one of claims 32 to 60,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:

the third message (M3) contains at least one optional second data field (dat2).

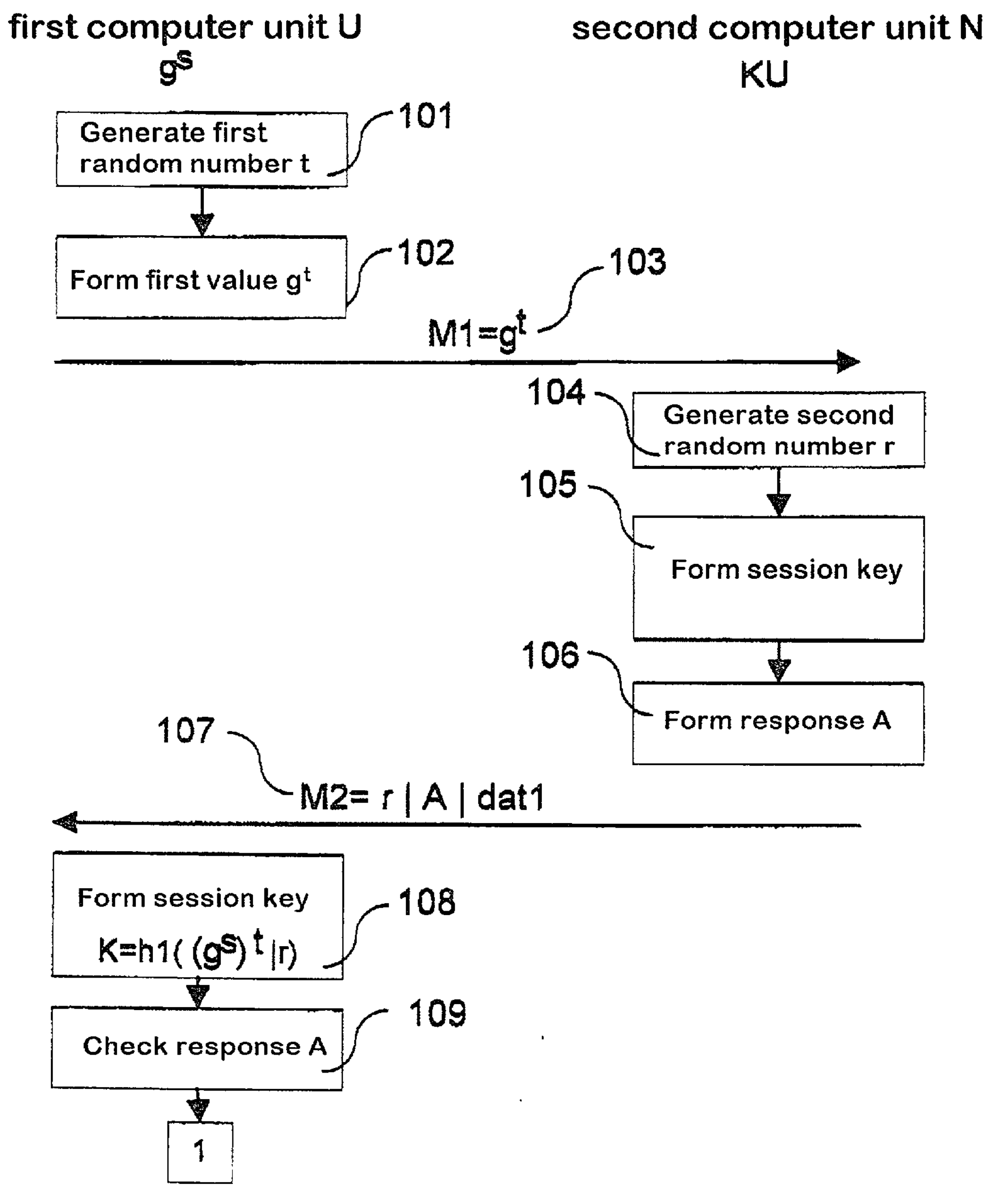
62. The arrangement as claimed in one of claims 32 to 61,
in which the first computer unit (U) and the second computer unit (N) are set up such that the following method steps can be carried out:
the first computer unit (U) is formed by a mobile communication terminal and/or the second computer unit (N) is formed by an authentication unit in a mobile communication network.

Abstract

Method and arrangement for the computer-aided interchange of cryptographic keys between a first computer unit and a second computer unit

The invention relates to a method which can be used to declare a session key (K) between a first computer unit (U) and a second computer unit (N) without it being possible for an unauthorized third party to obtain useful information regarding the keys or the identity of the first computer unit (U). This is achieved by embedding the principle of El-Gamal key interchange in the method with additional formation of a digital signature using a hash value whose input variable contains at least variables which can be used to infer the session key unambiguously.

FIG 1



first computer unit U
g^s
second computer unit N
KU
Generate first random number t
101
Form first value g^t
102
103
M1=g^t
104
Generate second random number r
105
Form session key
106
Form response A
107
M2= r | A | dat1
Form session key
K=h1((g^s)^t |r)
108
Check response A
109
1

FIG 1

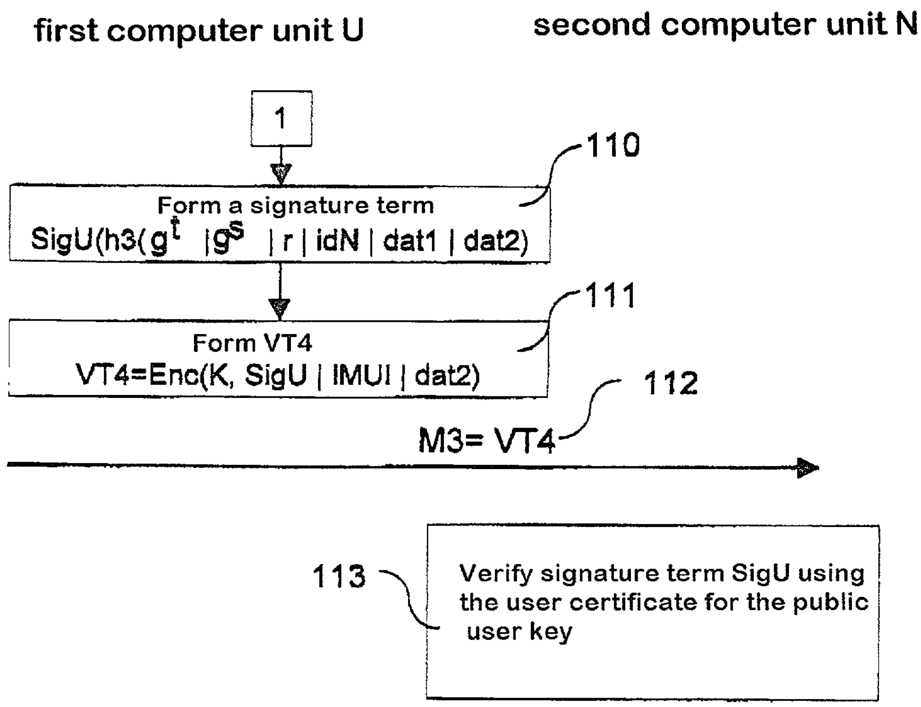

first computer unit U
second computer unit N
1
110
Form a signature term
SigU(h3(g^t | g^s | r | idN | dat1 | dat2)
111
Form VT4
VT4=Enc(K, SigU | IMUI | dat2)
112
M3= VT4
113
Verify signature term SigU using the user certificate for the public user key

FIG 2

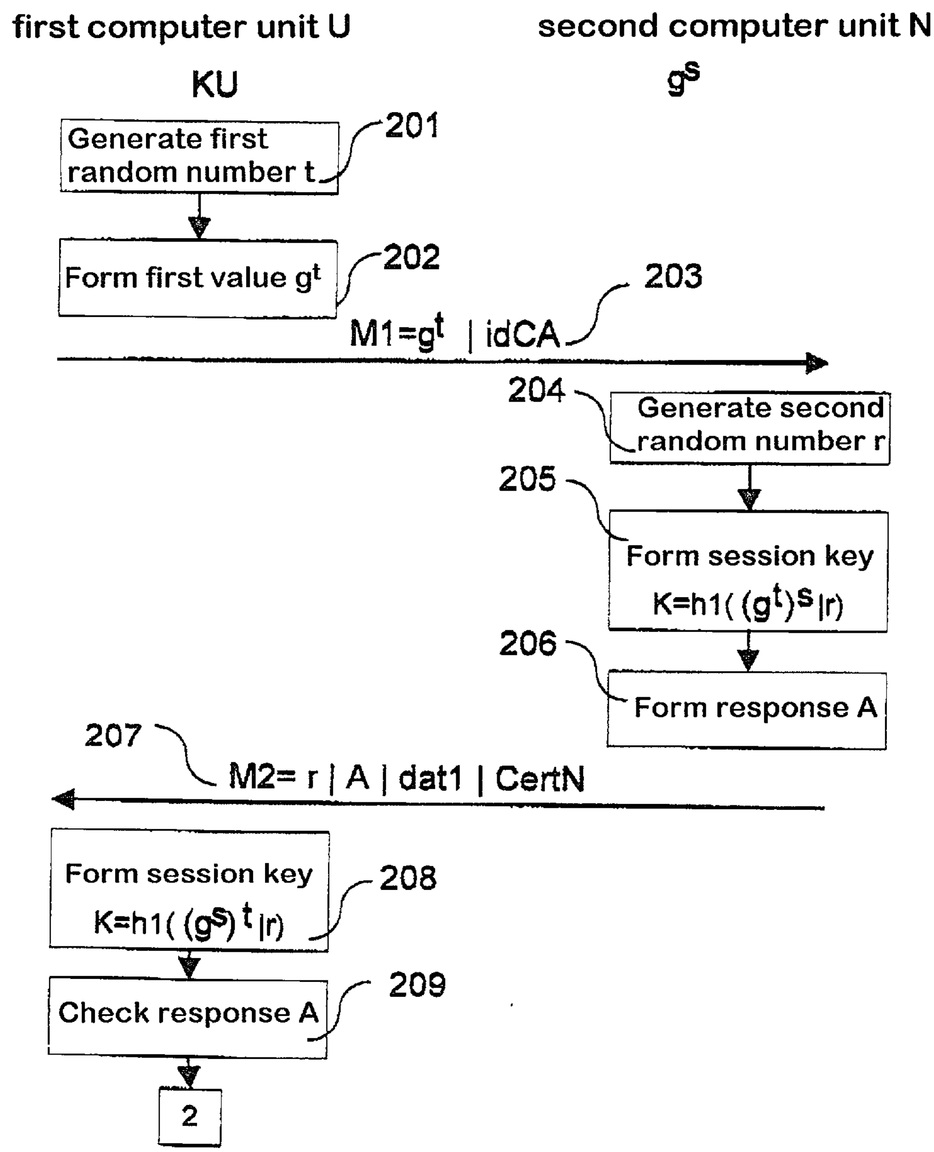

first computer unit U
second computer unit N
KU
g^s
Generate first random number t
201
Form first value g^t
202
203
M1=g^t | idCA
204
Generate second random number r
205
Form session key K=h1((g^t)^s |r)
206
Form response A
207
M2= r | A | dat1 | CertN
Form session key K=h1((g^s) ^t |r)
208
209
Check response A
2

FIG 2

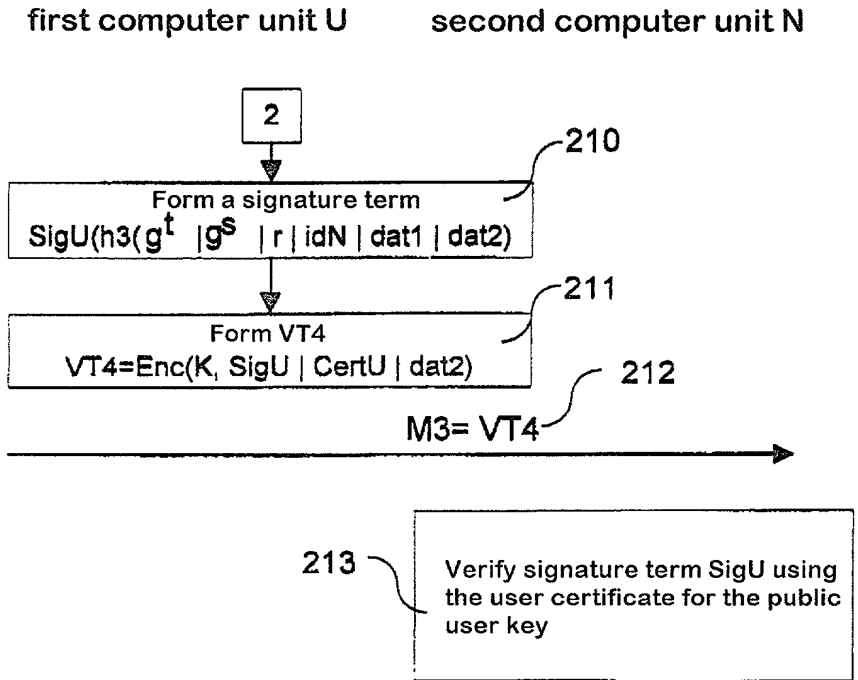

first computer unit U
second computer unit N
2
210
Form a signature term
SigU(h3(g^t |g^s | r | idN | dat1 | dat2)
211
Form VT4
VT4=Enc(K, SigU | CertU | dat2)
212
M3= VT4
213
Verify signature term SigU using the user certificate for the public user key

FIG 3

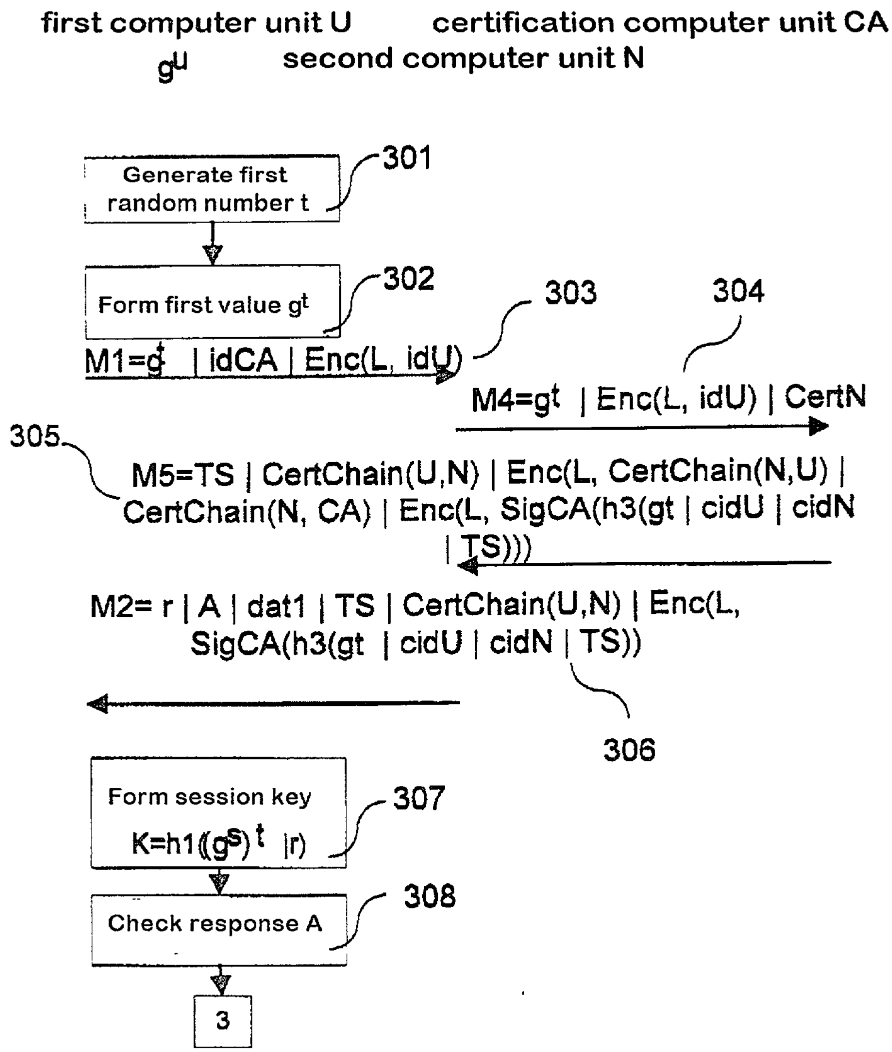
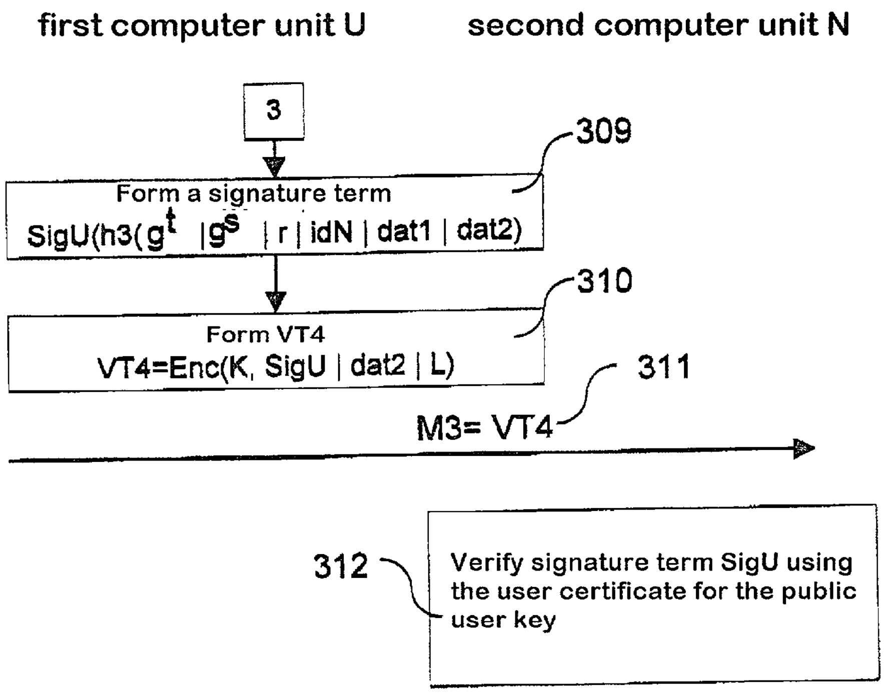

first computer unit U
certification computer unit CA
g^U
second computer unit N
Generate first random number t
301
Form first value g^t
302
M1=g^t | idCA | Enc(L, idU)
303
304
M4=g^t | Enc(L, idU) | CertN
305
M5=TS | CertChain(U,N) | Enc(L, CertChain(N,U) | CertChain(N, CA) | Enc(L, SigCA(h3(gt | cidU | cidN | TS)))
M2= r | A | dat1 | TS | CertChain(U,N) | Enc(L, SigCA(h3(gt | cidU | cidN | TS))
306
Form session key K=h1((g^s)^t |r)
307
Check response A
308
3

FIG 3


first computer unit U
second computer unit N
3
309
Form a signature term
SigU(h3(g^t | g^s | r | idN | dat1 | dat2)
310
Form VT4
VT4=Enc(K, SigU | dat2 | L)
311
M3= VT4
312
Verify signature term SigU using the user certificate for the public user key

09/700928

Docket No.: GR 98 P 1764 P

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	:	Guenther Horn et al.
Appl. No.	:	09/700,928
PCT No.	:	PCT/DE99/01365
Filed	:	November 20, 2000
Title	:	Method and Arrangement for the Computer-Aided Exchange of Cryptographic Keys Between a First Computer Unit and a Second Computer Unit
Art Unit	:	

ASSOCIATE POWER OF ATTORNEY

Hon. Commissioner of Patents and Trademarks,
Washington, D.C. 20231

Sir:

(2) Please recognize GREGORY L. MAYBACK (Reg. No. 40,719) as my associate in the matter in the above-identified application, with full powers. Please continue addressing all communications to the following address:

Lerner and Greenberg, P.A.
P.O. Box 2480
Hollywood, Florida 33022-2480

Respectfully submitted,

For Applicant(s)

LAURENCE A. GREENBERG
REG. NO. 29,308

Date: March 26, 2001

Lerner and Greenberg, P.A.
Post Office Box 2480
Hollywood, FL 33022-2480
Tel: (954) 925-1100
Fax: (954) 925-1101

/mjb

Practitioner's Docket No. GR 98 P 1764 **_PATENT_**

COMBINED DECLARATION AND POWER OF ATTORNEY

(ORIGINAL, DESIGN, NATIONAL STAGE OF PCT, SUPPLEMENTAL, DIVISIONAL, CONTINUATION, OR C-I-P)

As a below named inventor, I hereby declare that:

TYPE OF DECLARATION

This declaration is for a national stage of PCT application.

INVENTORSHIP IDENTIFICATION

My residence, post office address and citizenship are as stated below, next to my name. I believe that I am the original, first and sole inventor *(if only one name is listed below)* or an original, first and joint inventor *(if plural names are listed below)* of the subject matter that is claimed, and for which a patent is sought on the invention entitled:

TITLE OF INVENTION

METHOD AND ARRANGEMENT FOR THE COMPUTER-AIDED EXCHANGE OF CRYPTOGRAPHIC KEYS BETWEEN A FIRST COMPUTER UNIT AND A SECOND COMPUTER UNIT

SPECIFICATION IDENTIFICATION

The specification is attached hereto.

ACKNOWLEDGMENT OF REVIEW OF PAPERS AND DUTY OF CANDOR

I hereby state that I have reviewed and understand the contents of the above-identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information, which is material topatentability as defined in 37, Code of Federal Regulations, § 1.56.

PRIORITY CLAIM (35 U.S.C. § 119(a)-(d))

I hereby claim foreign priority benefits under Title 35, United States Code, § 119(a)-(d) of any foreign application(s) for patent or inventor's certificate or of any PCT international application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign application(s) for patent or inventor's

certificate or any PCT international application(s) designating at least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed.

Such applications have been filed as follows.

PRIOR PCT APPLICATION(S) FILED WITHIN 12 MONTHS (6 MONTHS FOR DESIGN) PRIOR TO THIS APPLICATION AND ANY PRIORITY CLAIMS UNDER 35 U.S.C. § 119(a)-(d)

INDICATE IF PCT	APPLICATION NUMBER	DATE OF FILING DAY, **MONTH**, YEAR	PRIORITY CLAIMED UNDER 35 USC 119
PCT	PCT/DE99/01365	06/05/99	
	198 22 795.7	20/05/98	yes

POWER OF ATTORNEY

I hereby appoint the following practitioner(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

Herbert L. Lerner	Registration Number 20435
Laurence A. Greenberg	Registration Number 29308
Werner H. Stemer	Registration Number 34,956
Ralph E. Locher	Registration Number 41,947

SEND CORRESPONDENCE TO

Lerner and Greenberg, P.A.
P.O. Box 2480
Hollywood, FL 33020-480
Tel.: (954) 925-1100
Fax: (954) 925-1101

DECLARATION

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

SIGNATURE(S)

GUENTHER HORN
Volker Kessler
Klaus Mueller

SIGNATURE(S)

1-00

GUENTHER HORN

Inventor's signature x Günther Horn

Date x 2000-05-12 Country of Citizenship GERMANY

Residence MUENCHEN, GERMANY DEX

Post Office Address EDUARD-SCHMID-STRASSE 16
D-81541 MUENCHEN
GERMANY

2-00

VOLKER KESSLER

Inventor's signature x Volker Kessler

Date x 2000-06-12 Country of Citizenship GERMANY

Residence GUMMERSBACH, GERMANY DEX

Post Office Address FURTWAENGLERSTRASSE 10
D-51643 GUMMERSBACH
GERMANY

3-00

KLAUS MUELLER

Inventor's signature x Klaus Müller

Date x 2000-05-12 Country of Citizenship GERMANY

Residence MUENCHEN, GERMANY DEX

Post Office Address RAINTALER STRASSE 15
D-81539 MUENCHEN
GERMANY